A Parasitics Extraction and Network Reduction Algorithm for Analog VLSI

Teng-Sin Pong and Martin A. Brooke, *Member, IEEE*

Abstract—This paper discusses an algorithm for the extraction of circuit parasitics in integrated circuits using classical transmission line models. This gives a better account of the dc and ac characteristics of interconnects than models incorporating exclusively either the R or C components. We also detail a network reduction technique used to simplify the extracted RC network at user specified accuracies to manageable complexities, especially for large VLSI circuits. The model and circuit reduction algorithms are applied to practical sample circuits. Results of simulations illustrating the reduction in circuit complexity and the degree of modeling accuracy by these methods are also given.

I. INTRODUCTION

WITH THE advancement of VLSI technologies, circuit designers require more accurate prediction of the effect of integrated circuit parasitics, such as gate resistances and capacitances [1]. In particular, analog circuits constructed in a short channel MOS technology require accurate prediction of parasitics to prevent mismatch, undesirable crosstalk, and uncontrolled effects at sensitive nodes.

Various papers have been published on this topic. Certain algorithms [2]-[7] involve calculating interconnect resistances using general Laplace equations, conformal transformations, finite elements and differences, or geometric heuristics. They all present varying degrees of accuracy and speed of calculation. Others [8]-[13] include parasitic capacitance calculation in their algorithms as well. However, all these algorithms consider the R and C of interconnect parasitics separately and independently, which is not strictly correct.

In [14], a model is presented which considers interconnects as a RC transmission line. The model gives accurate simulation results for simple lines but provides no solution to the complexity problem encountered when it is applied to large-scale networks because of their higher order.

In the extraction of complex VLSI circuits, after accurately modeling the parasitics, a network reduction algorithm is necessary to simplify them. There should also be a way to control the accuracy of the simplified network at user specified frequencies. This will ensure that no unnecessary accuracy is used in the model when it is uncalled for.

This paper presents a transmission line model for parasitics extraction, coupled with a network simplification algorithm in order to work efficiently even with big VLSI circuit layouts. Extraction and network reduction are carried out at accuracies specified by the operator.

II. MODELING OF TRANSMISSION LINE INTERCONNECTS

It is necessary to use an accurate model to describe circuit parasitics. From transmission line theory, for a line of length D and at a given frequency ω, exact equivalent T and Pi networks can be found. These networks have the form shown in Fig. 1.

In Fig. 1, D = length of the line, Z_0 = $\sqrt{(R_0 + j\omega L_0 / G_0 + j\omega C_0)}$ is the characteristic impedance of the line, $\gamma = \sqrt{(R_0 + j\omega L_0)(G_0 + j\omega C_0)} = \alpha + j\beta$ is the propagation function, R_0, L_0, G_0, and C_0 are the resistance, inductance, conductance, and capacitance per unit length, respectively.

We notice that, for $|\gamma D| \ll 1$:

$$\sinh \gamma D \simeq \gamma D \tag{1}$$

$$\cosh \gamma D \simeq 1 \tag{2}$$

$$\tanh \frac{\gamma D}{2} \simeq \frac{\gamma D}{2}. \tag{3}$$

So

$$Z_0 \times \tanh \frac{\gamma D}{2} \simeq \frac{(R_0 + j\omega L_0) \times D}{2} \tag{4}$$

$$\frac{Z_0}{\sinh \gamma D} \simeq \frac{1}{(G_0 + j\omega C_0) \times D}. \tag{5}$$

We can, therefore, replace the T and Pi networks in Fig. 1 by simplified models, as illustrated in Fig. 2.

These simplified models can further be approximated by the use of only R and C elements, i.e., by neglecting the L and G components. Such approximations will incur an error of approximation, *global error* (D), which can be calculated as follows.

In analog VLSI, $\omega L_0 \ll R_0$ and $G_0 \ll \omega C_0$ are generally true (a set of typical values are: $\omega = 2\pi \times 100$ MHz, $L_0 = 10^{-15}$ H/μm, $R_0 = 0.01$ Ω/square, $G_0 = 10^{-12}$ Ω^{-1} square, $C_0 = 10^{-16}$ F/μm^2). Then:

$$Z_0 \simeq \sqrt{\frac{R}{j\omega c_0}} = \sqrt{\frac{R_0}{\omega C_0}} \times e^{(-j\pi/4)} \tag{6}$$

$$\gamma \simeq \sqrt{j\omega R_0 C_0} = \sqrt{\omega R_0 C_0} \times e^{(j\pi/4)} \tag{7}$$

$$= \sqrt{\frac{\omega R_0 C_0}{2}} \times (1 + j) = \alpha + j\beta. \tag{8}$$

Then:

$$\alpha = \beta = \sqrt{\frac{\omega R_0 C_0}{2}} \tag{9}$$

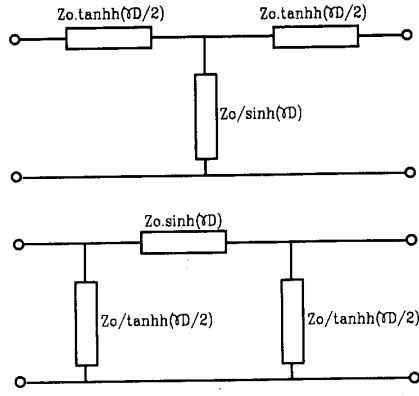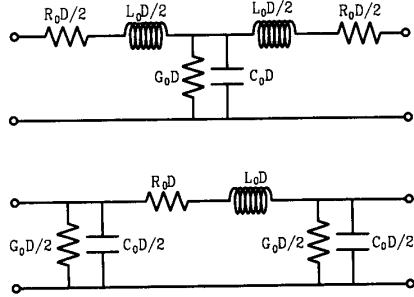$$\sinh \gamma D = \sinh \beta D \cos \beta D + j \cosh \beta D \sin \beta D. \tag{10}$$

Fig. 1. Exact T and Pi transmission line models.

Fig. 2. Simplified transmission line models.

The error when approximating $\sinh \gamma D \simeq \gamma D = \alpha D + j\beta D$ is

$$\text{error}(R) = 1 - \frac{\beta D}{\sinh \beta D \cos \beta D} \quad (11)$$

$$\text{error}(Im) = 1 - \frac{\beta D}{\cosh \beta D \sin \beta D} \quad (12)$$

where error (R) and error (Im) are errors in the real and imaginary parts of the approximation, respectively. We then define

$$\text{global error}(D) = \max\left(\text{error}(R), \text{error}(Im)\right). \quad (13)$$

Since $\tanh \gamma D = (\sinh \gamma D / \cosh \gamma D)$, and $\cosh x \simeq 1$ is a much better approximation than $\sinh x \simeq x$ (gradient of $\cosh$ is much smaller than $\sinh$ at the origin), the error of $\tanh \gamma D \simeq \gamma D$ is lower than that of $\sinh \gamma D \simeq \gamma D$. The error of approximating $\tanh (\gamma D/2) \simeq (\gamma D/2)$ is, therefore, limited by the expression of global error (D) derived for $\sinh \gamma D \simeq \gamma D$ above.

To give an example, let us take the polysilicon transmission line in SCMOS 2.0-μm technology [15] which has: $R_0 = 25.3$ Ω/square, $C_0 = 39$ aF/μm^2. Then, at $\omega = 2\pi \times 100$ MHz, $\beta = 5.60 \times 10^{-4}$ rad/μm. At these values:

For $D = 50$ μm, global error $(D) = 0.5\%$

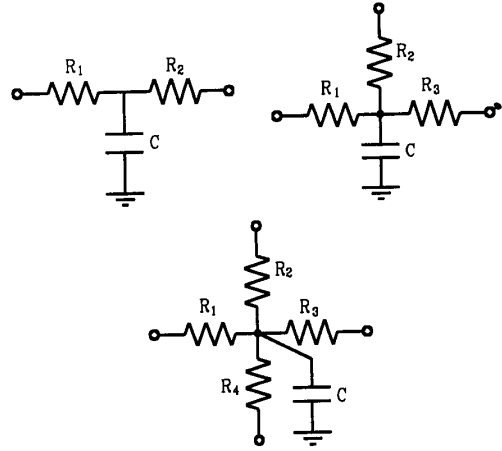For $D = 500$ μm, global error $(D) = 4.99\%$.

Fig. 3. Models for multipronged line elements.

In fact, it is found that, in order to satisfy an error threshold of 1% on the model, polysilicon lines should not exceed the length of 200 μm. For the p-diffusion layer, using the values of $R_0 = 130.4$ Ω/square, $C_0 = 270$ atF/μm^2, and $\beta = 3.33 \times 10^{-3}$ rad/μm, line dimensions must not exceed 37 μm.

With modern VLSI circuits, certain feature sizes can easily exceed 3000 μm, which is half the size of a 250-mil chip. The values of D calculated previously then indicate that any feature whose size is bigger than the maximum allowable D value should be cut into smaller pieces in order to allow accurate modeling at the accuracy chosen by the user. A way to do this is to calculate max (D) values for each different material and technology and for each stated error tolerance level. Subsequent modeling using line segments shorter than max (D) will then ensure that errors incurred by using simplified T or Pi equivalent networks shall be within manageable proportions.

In our extraction algorithm, we will, therefore, limit the maximum length of elementary transmission lines to max (D), calculated for each of the layers used in the technology for power lines and interconnects. Long lines will thus be segmented into shorter lengths in order to satisfy the error tolerance specified for the modeling.

As is shown in [13], both the Pi- and T-type transmission line models are accurate. There is no particular advantage of one of them over the other. However, in our computer extraction program, the T model is used because it makes subsequent network reduction easier.

From the T model we derive starred configurations for multiple pronged line elements. We need these variations because circuit connectivity in Manhattan designs can generate line segments connected to up to 4 other neighbors. A few examples appear in Fig. 3.

For an n-pronged line element, current paths are assumed to flow from each of the connected edges towards the geometric center of the rectangle. Each current path, therefore, sees half of the rectangle in question in terms of length (in the direction of the current) and width (perpendicular to the direction of current). The corresponding R and C values are then calculated from these assumptions. In addition, in order to account for fringing capacitance effects, edges bordering on space or wells
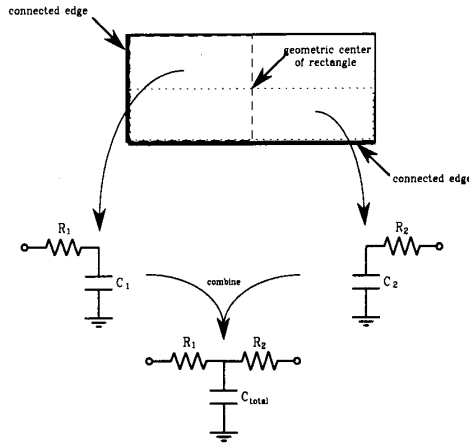
Fig. 4. Example of the derivation of a two-pronged line element model.

Fig. 5. Star-delta conversion example: notice the center node being eliminated during the conversion.

will have their capacitance values adjusted according to measured values for the particular technology in use. The correction is linear in terms of the length of the edge in question. Admittedly this approximation is not as accurate as methods such as the Laplace equations, but is sufficient for the purpose of providing a simple enough circuit that predicts the influence of integrated circuit parasitics on analog circuit performance to within a given error tolerance level. An example for the derivation of a two-pronged line is given in Fig. 4.

III. RC NETWORK REDUCTION

The transmission line model provides us with an accurate way to calculate circuit parasitics. However, the network thus extracted from actual VLSI circuits would contain too many branches to be practical for any simulation purposes. So there is a need to be able to simplify the extracted network, preferably at user controlled accuracy, as well.

Using classical network theories, we know that for any star configuration, there exists an equivalent delta configuration with one less node.

In Fig. 5, we have

$$Z_{ij} = z_i \times z_j \times \sum_{k=1}^{n} \frac{1}{z_k} = \text{Re}\,(Z_{ij}) + j\,\text{Im}\,(Z_{ij}). \quad (14)$$

This simple formula gives an *exact* transformation of star to delta networks. Here, the reader's attention is called to the fact that the technique works for the frequency domain only. For our network reduction algorithm, we will recursively employ this star-delta conversion to simplify the extracted circuit. After each star-delta conversion, we will try to simplify the resulting delta network to one consisting of only purely resistive and purely capacitive branches. This is done by simply neglecting the smaller (in terms of admittance) component of the real or imaginary parts.

An example dealing with a symmetrical 4-pronged line with only R and C branches is given in Fig. 6.

For a branch simplified to a resistor

$$\text{error}\,(R_{\text{branch}}) = \left(\frac{\text{Im}\,(Z_{ij})}{\text{Re}\,(Z_{ij})} \right)_{R_{\text{branch}}}. \quad (15)$$
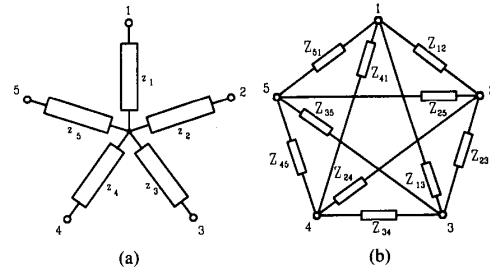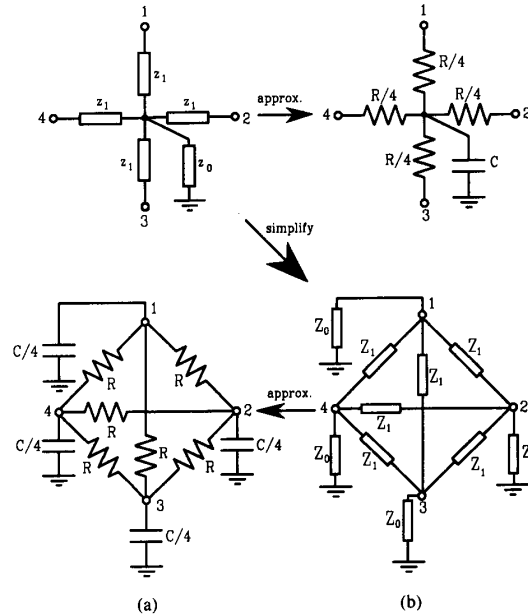

Fig. 6. Example: simplifying a multipronged line by star-delta conversion.

For a branch simplified to a capacitor

$$\text{error}\,(C_{\text{branch}}) = \left(\frac{\text{Re}\,(Z_{ij})}{\text{Im}\,(Z_{ij})} \right)_{C_{\text{branch}}}. \quad (16)$$

We then define the overall error, called *maximum error*, to be:

$$\text{maximum error} = \max \left(\max_{\text{all } R\text{branches}} \left(\text{error}(R_{\text{branch}}) \right) \right.$$

$$\left. \max_{\text{all } C\text{branches}} \left(\text{error}\,(C_{\text{branch}}) \right) \right). \quad (17)$$

When the maximum error exceeds a specified threshold value, the conversion and subsequent approximation to purely R and C branches will *not* be carried out. Notice that the complex impedance transmission line model of the interconnect is used as the starting point for all simplifications. This ensures that the errors calculated are the exact relative errors between the unsimplified and simplified interconnect models. This means that

any impedance measurement taken from the simplified inter-connect model will match a similar measurement taken from the unsimplified model to within the user specified error at the user specified frequency.

IV. ALGORITHM FOR EXTRACTION AND NETWORK REDUCTION

Using the transmission line and network simplification theo-ries as explained previously, we arrive at our parasitics extrac-tion algorithm for Manhattan designs.

a) Find all intersections of line elements and cut up the lines at the intersecting coordinates.

b) Given the working frequency and error tolerance for net-work reduction (these can be user inputs), calculate the maxi-mum line segment dimensions, for each layer material, to be extracted.

c) Using the values from b), find interconnect lines and other parasitic elements whose dimensions are too big and divide them into smaller pieces whose length and width are *both* below the maximum allowable feature size.

d) Extract connectivity information and number nodes ac-cordingly.

e) For each rectangle, generate line elements using the T transmission line model with exact *complex* impedance values as shown in Fig. 1 for each of its edges that is a connection node. Each line element's components are calculated for the half-rectangle that the current sees flowing from the edge to the geometric center of the full rectangle. Then connect the line element branches using connectivity information from d).

f) Simplify the network using the star–delta conversion for-mulas recursively on every node until the error threshold given by the user governing the reducibility of any star configuration prevents any further reduction.

V. IMPLEMENTATION AND RESULTS

The transmission line model and RC network reduction al-gorithm are implemented in a C language program of approxi-mately 2700 lines in source code. The program assumes the input to be in the Magic layout file format and produces a SPICE simulation deck as output.

The program was run on the Magic layout file of several sim-ple layouts, whose responses are known. Parameters such as the error threshold and working frequency were varied to compare their influence on the accuracy of the extracted circuit. The fol-lowing are the results from the extraction of two typical layouts. In both cases, the transmission line model was calculated at an error tolerance level of 10%, with the working frequency at 100 MHz. The network reduction algorithm was carried out up to a 10% error threshold. The accuracy of the reduced network is compared to the unreduced model, extracted at 10% accuracy, in terms of input node impedance as seen from outside the cir-cuit at specific nodes, and the relative percentage errors are plotted. This type of comparison is suitable for the prediction of signal behavior on a transmission line that goes on an I/O node to drive a transistor, for example.

Example #1 (Fig. 7) shows how a complex RC network is effectively reduced to the minimum allowed by the star–delta conversion method, i.e., $(n \times (n - 1)/2)$ branches for n I/O nodes. The number of RC branches and nodes was thus reduced by a factor of about 4–7. This example layout is typical of power or clock lines with numerous distribution points. When the star–delta conversion/reduction process stops because the error
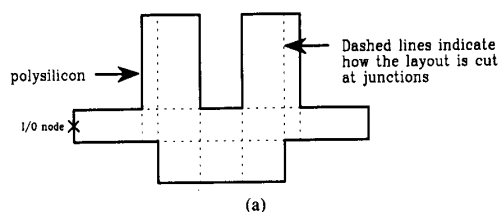
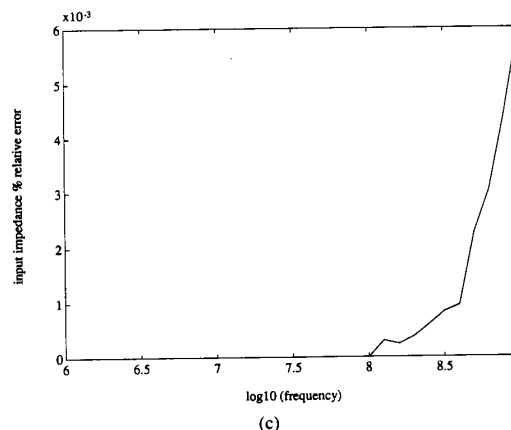| # of RC branches before reduction/ after reduction	# of nodes before reduction/ after reduction
51/1	32/1

(b)

Fig. 7. Example #1. (a) The layout shape; layout: butting junction. (b) Circuit complexity before and after reduction; reduction in circuit com-plexity and accuracy. (c) Accuracy of reduced circuit measured in terms of relative percentage error in input impedance at node marked x.

threshold has been reached, a further reduction of the circuit can also be envisaged by combining I/O ports interconnected by impedances which are small enough to be considered as practical short circuits. In such cases, the test that ensures the accuracy of the process is tied in with the sensitivity of the sim-ulator accuracy and user specified voltage and current accura-cies.

Example #2 (Fig. 8) illustrates the effect of different char-acteristic impedance values of a layer type on the complexity and accuracy of the extracted SPICE file (i.e., the R and C were increased 10 fold arbitrarily to simulate the change in layer and characteristic impedance).

While the two frequency plots above are given for the range of frequencies from 1 MHz to 1 GHz, the part of the curves beyond 100 MHz are not to be taken as literally predicting ex-cellent performance at microwave frequency ranges. What the graphs do predict is that the reduced networks accurately rep-resent the behavior of transmission lines at least up till the working frequency (or frequencies) at which the network was extracted and reduced, and this to within a user specified error tolerance level.

From these examples and others, it is noted that, at relatively high error thresholds for the reduction algorithm (up to even 50% in simple cases), the extracted circuits still present re-markable accuracy at frequencies up to the working frequency. This indicates that error thresholds can be relaxed quite gener-ously in lieu of more simplicity in the extracted circuit and sim-ulation speed.

(a)

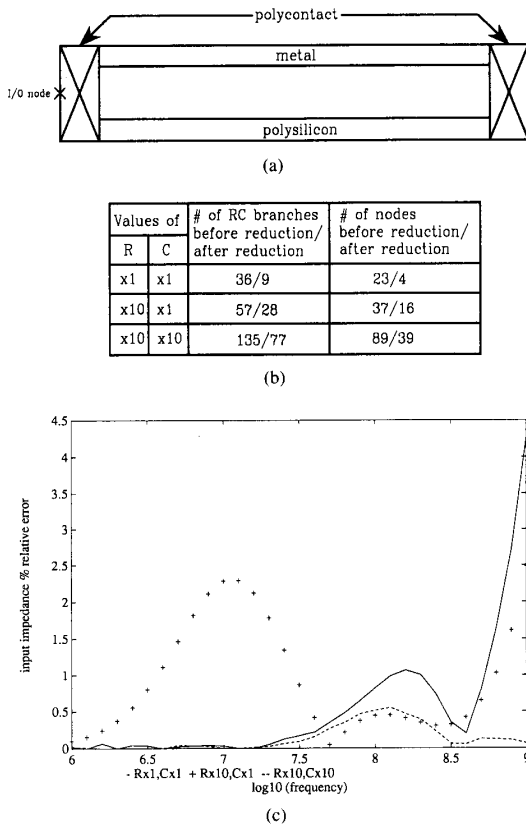| Values of		# of RC branches before reduction/ after reduction	# of nodes before reduction/ after reduction
R	C		
x1	x1	36/9	23/4
x10	x1	57/28	37/16
x10	x10	135/77	89/39

(b)

(c)

Fig. 8. Example #2: Effect of characteristic impedances. (a) The layout shape: parallel polymetal lines. (b) Circuit complexity before and after reduction; reduction in circuit complexity and accuracy. (c) Accuracy of reduced circuit measured in terms of relative percentage error in input impedance at node marked x.

VI. CONCLUSION

The transmission line model can be used effectively and accurately in the extraction of parasitics, and affords a better account of the effects of parasitics in analog integrated circuits. The network reduction technique presented is attractive in that it can accommodate a compromise between speed and accuracy during simulation and produce simulation parameters accordingly.

Although this algorithm is written in the Magic context, the general approach in applying transmission line theory and network simplification to parasitics extraction can be extended to other CAD layout data bases with the Manhattan design rule.

REFERENCES

[1] D. L. Carter and D. F. Guise, "Effects of interconnections on submicron chip performance," *VLSI Design*, pp. 63-66, Jan. 1984.

[2] B. R. Chawla and H. K. Gummel, "A boundary technique for calculation of distributed resistance," *IEEE Trans. Electron. Devices*, vol. ED-17, pp. 915-925, Oct. 1970.

[3] T. Mitsuhashi and K. Yoshida, "A resistance calculation algorithm and its application to circuit extraction," *IEEE Trans. Computer-Aided Design*, vol. CAD-6, pp. 337-345, May 1987.

[4] M. Horowitz and R. W. Dutton, "Resistance extraction from mask layout data," *IEEE Trans. Computer-Aided Design*, vol. CAD-2, pp. 145-150, July 1983.

[5] H. Yoshimura et al., "An algorithm for resistance calculation from IC mask pattern recognition," in *Proc. 1979 ISCAS*, 1979, pp. 478-481.

[6] M. G. Harbour and J. M. Drake, "Calculation of multiterminal resistances in integrated circuits," *IEEE Trans. Circuits Syst.*, vol. CAS-33, pp. 462-465, Apr. 1986.

[7] A. J. Kemp, J. A. Pretorius, and W. Smit, "Generation of a mesh for resistance calculation in integrated circuits," *IEEE Trans. Computer-Aided Design*, vol. 7, pp. 1029-1037, Oct. 1988.

[8] W. C. Elmore, "Transient response of damped linear networks with particular regard to wideband amplifiers," *IEEE J. Appl. Phys.*, vol. 19, pp. 55-63, Jan. 1948.

[9] S.-Lin Su, V. B. Rao, and T. N. Trick, "A simple and accurate node reduction technique for interconnect modeling in circuit extraction," in *Proc. IEEE Int. Conf. on CAD: ICCAD-86*, 1986, pp. 270-273.

[10] J. Rubinstein, P. Penfield, Jr., and M. A. Horowitz, "Signal delay in RC tree networks," *IEEE Trans. Computer-Aided Design*, vol. CAD-2, pp. 202-210, July 1983.

[11] J. L. Wyatt, "Signal delay in RC mesh networks," *IEEE Trans. Circuits Syst.*, vol. CAS-32, pp. 507-510, May 1985.

[12] W. D. Smith, J. McDonald, C.-T. Chang, and R. Jerdonek, "Next: A hierarchical layout verification system for VLSI," *IEEE Int. Conf. on Computer Design: VLSI in Computers ICCD '84-Proc.*, 1984, pp. 820-825.

[13] T. Sakurai, "Approximation of wiring delay in MOSFET LSI," *IEEE J. Solid-State Circuits*, vol. SC-18, pp. 418-426, Aug. 1983.

[14] R. J. Antinone and G. W. Brown, "The modeling of resistive interconnects for integrated circuits," *IEEE J. of Solid-State Circuits*, vol. SC-8, pp. 200-203, Apr. 1983.

[15] MOSIS, Parameters for batch M96L.

Teng-Sin Pong received the Diplôme d'Ingenieur from L'Institut Superieur d'Electronique de Paris, Paris, France, in 1987 and the M.S.E.E. degree from the Georgia Institute of Technology in 1989. He is currently working toward the Ph.D. degree at the Georgia Institute of Technology.

His research interests are in various aspects of computer-aided integrated circuit design, with particular emphasis on circuit extraction and simulation tools for analog circuit designers.

Martin A. Brooke (M'90) received the B.E. degree from Auckland University in New Zealand in 1981, and the M.S. and Ph.D. degrees in electrical engineering from the University of Southern California in 1984 and 1988, respectively.

He is currently employed as an Assistant Professor of Electrical Engineering at The Georgia Institute of Technology. His current research interests include: analog circuit design tools, in particular the modeling of electronic circuit and device parasitics, and the development of simulation models for electronic devices from process information; parallel analog circuit architectures for high-speed high precision analog signal processing, including high-speed A/D converter design; digitally controlled electrically reconfigurable analog circuits; and analog circuit neural network implementations.